A Jacobi Method for Lattice Basis Reduction

Sanzheng Qiao

Department of Computing and Software
McMaster University, Canada
qiao@mcmaster.ca
www.cas.mcmaster.ca/~qiao

January 30, CSE/SHARCNET

Joint work with Zhaofei Tian
Outline

1. Lattices, Bases, and Reduction
2. MIMO Wireless
3. Jacobi Method
4. Experimental Results
Outline

1. Lattices, Bases, and Reduction
2. MIMO Wireless
3. Jacobi Method
4. Experimental Results
A (point) lattice is a periodic arrangement of discrete points. The set

\[ L = \{Az \mid z \in \mathbb{Z}^n\} \]

is call the lattice generated by \( A \).

Basis: Formed by the columns of \( A \) (generator matrix).
Two-dimensional examples

Regular square lattice,

\[ A = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \]
Two-dimensional examples

Rhombic lattice,

\[ A = \begin{bmatrix} 1 & 1/\sqrt{2} \\ 0 & 1/\sqrt{2} \end{bmatrix} \]
For a given lattice, its basis is not unique.

$$A = \begin{bmatrix} -1 & 4 \\ -2 & 3 \end{bmatrix}$$
Another basis for the same lattice.

\[ B = \begin{bmatrix} -1 & 2 \\ -2 & -1 \end{bmatrix} \]
Bases

Which basis, $A$ or $B$ is more preferable?
Which basis, $A$ or $B$ is more preferable?

$B$

- shorter
- “more orthogonal” to each other
Two bases are related by $AZ = B$ ($A = BZ^{-1}$).
Two bases are related by $AZ = B$ ($A = BZ^{-1}$).

$Z$: Unimodular matrix, a nonsingular integer matrix whose inverse is also integer. (An integer matrix whose determinant is ±1.)
Unimodular matrix

Two bases are related by \( AZ = B \ (A = BZ^{-1}) \).

\( Z \): Unimodular matrix, a nonsingular integer matrix whose inverse is also integer. (An integer matrix whose determinant is \( \pm 1 \).)

\[
\begin{bmatrix}
-1 & 4 \\
-2 & 3
\end{bmatrix}
\begin{bmatrix}
1 & 2 \\
0 & 1
\end{bmatrix}
= 
\begin{bmatrix}
-1 & 2 \\
-2 & -1
\end{bmatrix}
\]
Lattice basis reduction problem:
Given a basis for a lattice, find a basis consisting of short (more orthogonal) vectors.

Lattice basis reduction algorithm:
Given a basis matrix $A$, compute a unimodular matrix $Z$ that transforms $A$ into a new basis matrix $B = AZ$ whose column vectors (basis vectors) are short and “more orthogonal” to each other.
Notions of reduction

- Hermite reduction, 1850
- HKZ (Hermite-Korkine-Zolotarev) reduction, 1873.
- Minkowski reduction, 1891
- LLL (Lenstra-Lenstra-Lovász) reduction, 1982
Notions of reduction

- Hermite reduction, 1850
- HKZ (Hermite-Korkine-Zolotarev) reduction, 1873.
- Minkowski reduction, 1891
- LLL (Lenstra-Lenstra-Lovász) reduction, 1982

Minkowski reduction is optimal (shortest basis vectors), exponential complexity.
Notions of reduction

- Hermite reduction, 1850
- HKZ (Hermite-Korkine-Zolotarev) reduction, 1873.
- Minkowski reduction, 1891
- LLL (Lenstra-Lenstra-Lovász) reduction, 1982

Minkowski reduction is optimal (shortest basis vectors), exponential complexity.
LLL reduction is the fastest (polynomial) algorithm that produces good results in practice. Widely used in practice.
Outline

1. Lattices, Bases, and Reduction
2. MIMO Wireless
3. Jacobi Method
4. Experimental Results
MIMO system
Math model:

\[ b = Ax + n \]

- **A**: channel matrix, real
- **x**: transmitted signal, integer
- **n**: white noise, real
- **b**: received signal, real
Maximum likelihood Decoding

Maximum-likelihood (optimal) solution

\[
\min_{x \in \mathbb{Z}^n} \|Ax - b\|_2^2
\]
Maximum likelihood Decoding

Maximum-likelihood (optimal) solution

$$\min_{x \in \mathbb{Z}^n} \| Ax - b \|^2_2$$

An integer least squares problem.
Maximum likelihood Decoding

Maximum-likelihood (optimal) solution

$$\min_{x \in \mathbb{Z}^n} \|Ax - b\|_2^2$$

An integer least squares problem.

Non-polynomial problem.
Example

\[ A = \begin{bmatrix} -1 & 4 \\ -2 & 3 \end{bmatrix}, \quad b = \begin{bmatrix} -0.4 \\ 4 \end{bmatrix} \]
Example

\[ A = \begin{bmatrix} -1 & 4 \\ -2 & 3 \end{bmatrix}, \quad b = \begin{bmatrix} -0.4 \\ 4 \end{bmatrix} \]
Suboptimal decoding

A low complexity suboptimal method: Zero forcing.
Solve for the real solution, then round it to its nearest integer.
Suboptimal decoding

A low complexity suboptimal method: Zero forcing.
Solve for the real solution, then round it to its nearest integer.

\[ A^{-1}b = \begin{bmatrix} -3.44 \\ -0.96 \end{bmatrix} \rightarrow \begin{bmatrix} -3 \\ -1 \end{bmatrix} \]
Suboptimal decoding

A low complexity suboptimal method: Zero forcing. Solve for the real solution, then round it to its nearest integer.

\[ A^{-1}b = \begin{bmatrix} -3.44 \\ -0.96 \end{bmatrix} \rightarrow \begin{bmatrix} -3 \\ -1 \end{bmatrix} \]
Suboptimal decoding

A low complexity suboptimal method: Zero forcing. Solve for the real solution, then round it to its nearest integer.

\[ A^{-1}b = \begin{bmatrix} -3.44 \\ -0.96 \end{bmatrix} \rightarrow \begin{bmatrix} -3 \\ -1 \end{bmatrix} \]

Is this the ML solution?
Decoding

Try \( B \), reduced \( A \).

\[
B^{-1}b = \begin{bmatrix}
-1.52 \\
-0.96
\end{bmatrix} \rightarrow \begin{bmatrix}
-2 \\
-1
\end{bmatrix}
\]
Decoding

Try $B$, reduced $A$.

\[
B^{-1}b = \begin{bmatrix}
-1.52 \\
-0.96 \\
\end{bmatrix} \rightarrow \begin{bmatrix}
-2 \\
-1 \\
\end{bmatrix}
\]

A closer (closest) lattice point (1.077 vs 1.166).
Decoding

The performance (bit-error-rate) of a low complexity suboptimal decoding is highly related to the orthogonality of the coefficient matrix.
Decoding

The performance (bit-error-rate) of a low complexity suboptimal decoding is highly related to the orthogonality of the coefficient matrix.

Lattice reduction can improve the orthogonality of the coefficient matrix (shorten the the columns of the matrix)
Decoding

The performance (bit-error-rate) of a low complexity suboptimal decoding is highly related to the orthogonality of the coefficient matrix.

Lattice reduction can improve the orthogonality of the coefficient matrix (shorten the columns of the matrix)

Lattice reduction aided decoding.
Outline

1. Lattices, Bases, and Reduction
2. MIMO Wireless
3. Jacobi Method
4. Experimental Results
Lagrange algorithm

Two-dimensional lattice reduction workhorse

\[
\text{if } \|a_1\|_2 < \|a_2\|_2 \\
\text{swap } a_1 \text{ and } a_2; \\
\text{endif} \\
\text{repeat} \\
\quad q = \lfloor a_1^T a_2 / \|a_2\|_2^2 \rfloor; \\
\quad t = a_1; \\
\quad a_1 = a_2; \\
\quad a_2 = t - q \cdot a_2; \\
\text{until } \|a_1\|_2 \leq \|a_2\|_2
\]
In the end, Lagrange-reduced basis

\[ \| a_1 \|_2 \leq \| a_2 \|_2 \quad \text{and} \quad \frac{|a_1^T a_2|}{\| a_1 \|_2^2} < \frac{1}{2} \]

Thus

\[ \frac{|a_1^T a_2|}{\| a_1 \|_2 \| a_2 \|_2} \leq \frac{|a_1^T a_2|}{\| a_1 \|_2^2} < \frac{1}{2} \]

The angle between \( a_1 \) and \( a_2 \) is between \( 2\pi/3 \) and \( \pi/3 \).
In the end, Lagrange-reduced basis

\[ \|a_1\|_2 \leq \|a_2\|_2 \quad \text{and} \quad \frac{|a_1^T a_2|}{\|a_1\|_2^2} < \frac{1}{2} \]

Thus

\[ \frac{|a_1^T a_2|}{\|a_1\|_2 \|a_2\|_2} \leq \frac{|a_1^T a_2|}{\|a_1\|_2^2} < \frac{1}{2} \]

The angle between \(a_1\) and \(a_2\) is between \(2\pi/3\) and \(\pi/3\).

For two-dimensional lattices, a Lagrange-reduced basis is Minkowski (optimally) reduced, shortest basis vectors or closest to orthogonal.
Jacobi method

A Lagrange based Jacobi-like method for lattice reduction:

Apply Lagrange algorithm to every pair \((a_i, a_j), 1 \leq i < j \leq n\), of the columns of \(A\).
A Lagrange based Jacobi-like method for lattice reduction:

Apply Lagrange algorithm to every pair \((a_i, a_j), 1 \leq i < j \leq n\), of the columns of \(A\).

A lot of inner product \(a_i^T a_j\) computation.
A Lagrange based Jacobi-like method for lattice reduction:

Apply Lagrange algorithm to every pair \((a_i, a_j), 1 \leq i < j \leq n\), of the columns of \(A\).

A lot of inner product \(a_i^T a_j\) computation.

Form the Gram matrix \(G = A^T A\):

\[
    g_{ii} = \|a_i\|_2^2 \quad \text{and} \quad g_{ij} = a_i^T a_j
\]

Apply Lagrange algorithm to \(G\) to compute a unimodular matrix \(Z\) so that \(AZ\) is reduced.
Outline

1. Lattices, Bases, and Reduction
2. MIMO Wireless
3. Jacobi Method
4. Experimental Results
Termination

Criterion:

\[ g_{ii} \leq g_{jj} \quad \text{and} \quad \frac{|g_{ij}|}{g_{ii}} < \frac{1}{2}, \quad \text{for} \ 1 \leq i < j \leq n. \]
Termination

Criterion:

\[ g_{ii} \leq g_{jj} \quad \text{and} \quad \frac{|g_{ij}|}{g_{ii}} < \frac{1}{2}, \quad \text{for} \ 1 \leq i < j \leq n. \]

Random matrices
Cyclic-by-row
Average number of sweeps of ten matrices of the same size

<table>
<thead>
<tr>
<th>size</th>
<th>maximal</th>
<th>average</th>
</tr>
</thead>
<tbody>
<tr>
<td>50</td>
<td>8</td>
<td>6.4</td>
</tr>
<tr>
<td>100</td>
<td>8</td>
<td>6.5</td>
</tr>
<tr>
<td>200</td>
<td>8</td>
<td>6.2</td>
</tr>
</tbody>
</table>
Running time

Random matrices
Average CPU time of ten matrices of the same size
In LLL algorithm, $\omega = 0.99$

<table>
<thead>
<tr>
<th>size</th>
<th>Jacobi</th>
<th>LLL</th>
</tr>
</thead>
<tbody>
<tr>
<td>50</td>
<td>0.040</td>
<td>0.229</td>
</tr>
<tr>
<td>100</td>
<td>0.147</td>
<td>0.528</td>
</tr>
<tr>
<td>200</td>
<td>0.432</td>
<td>3.159</td>
</tr>
</tbody>
</table>
Orthogonality

Orthogonality defect

\[ \delta(A) = \left( \frac{\prod_j \|a_j\|_2}{|\det(A)|} \right)^{1/n} \]

Scaled geometric mean of column lengths, since

\[ |\det(A)| = |\det(AZ)| \]

for any unimodular \( Z \).
Orthogonality

Orthogonality defect

$$\delta(A) = \left( \frac{\prod_j \|a_j\|_2}{|\det(A)|} \right)^{1/n}.$$  

Scaled geometric mean of column lengths, since $|\det(A)| = |\det(AZ)|$ for any unimodular $Z$.

Random matrices

Average of ten matrices of the same size

In LLL algorithm, $\omega = 0.99$

<table>
<thead>
<tr>
<th>size</th>
<th>$\delta(A)$</th>
<th>Jacobi</th>
<th>LLL</th>
</tr>
</thead>
<tbody>
<tr>
<td>50</td>
<td>3.0230</td>
<td>1.9959</td>
<td>2.0578</td>
</tr>
<tr>
<td>100</td>
<td>3.1560</td>
<td>2.0902</td>
<td>2.3705</td>
</tr>
<tr>
<td>200</td>
<td>3.2370</td>
<td>2.1638</td>
<td>2.4046</td>
</tr>
</tbody>
</table>
Recent work

Proved convergence.
Recent work

Proved convergence.

Enhanced Jacobi method:
Integrating the Hermite (size) reduction into the Jacobi method to improve the orthogonality of the columns of the reduced matrix, measured by both widely used condition number and orthogonality defect.
Enhanced Jacobi method

Compared with the LLL algorithm.

Condition number
Enhanced Jacobi method

Orthogonality defect
Enhanced Jacobi method

Speed
Thank you!
Thank you!

Questions?